

Lab Exercises

Name: _____

Lab Exercise 3 — Creating an Employee Class

Solution

```

1 // Lab 3: Employee.h
2 // Employee class definition.
3
4 #include <string> // program uses C++ standard string class
5 using namespace std;
6
7 // Employee class definition
8 class Employee
9 {
10 public:
11     Employee( string, string, int ); // constructor sets data members
12     void setFirstName( string ); // set first name
13     string getFirstName(); // return first name
14     void setLastName( string ); // set last name
15     string getLastName(); // return last name
16     void setMonthlySalary( int ); // set weekly salary
17     int getMonthlySalary(); // return weekly salary
18 private:
19     string firstName; // Employee's first name
20     string lastName; // Employee's last name
21     int monthlySalary; // Employee's salary per month
22 }; // end class Employee

```

```

1 // Lab 3: Employee.cpp
2 // Employee class member-function definitions.
3 #include <iostream>
4 using namespace std;
5
6 #include "Employee.h" // Employee class definition
7
8 // Employee constructor initializes the three data members
9 Employee::Employee( string first, string last, int salary )
10 {
11     setFirstName( first ); // store first name
12     setLastName( last ); // store last name
13     setMonthlySalary( salary ); // validate and store monthly salary
14 } // end Employee constructor
15
16 // set first name
17 void Employee::setFirstName( string name )
18 {
19     firstName = name; // no validation needed
20 } // end function setFirstName
21
22 // return first name
23 string Employee::getFirstName()
24 {
25     return firstName;
26 } // end function getFirstName
27
28 // set last name
29 void Employee::setLastName( string name )
30 {
31     lastName = name; // no validation needed
32 } // end function setLastName

```

Lab Exercises

Name: _____

Lab Exercise 3 — Creating an Employee Class

```

33
34 // return last name
35 string Employee::getLastName()
36 {
37     return lastName;
38 } // end function getLastName
39
40 // set monthly salary; if not positive, set to 0.0
41 void Employee::setMonthlySalary( int salary )
42 {
43     if ( salary > 0 ) // if salary is positive
44         monthlySalary = salary; // set monthlySalary to salary
45
46     if ( salary <= 0 ) // if salary is not positive
47         monthlySalary = 0; // set monthlySalary to 0.0
48 } // end function setMonthlySalary
49
50 // return monthly salary
51 int Employee::getMonthlySalary()
52 {
53     return monthlySalary;
54 } // end function getMonthlySalary

```

```

1 // Lab 3: EmployeeTest.cpp
2 // Create and manipulate two Employee objects.
3 #include <iostream>
4 using namespace std;
5
6 #include "Employee.h" // include definition of class Employee
7
8 // function main begins program execution
9 int main()
10 {
11     // create two Employee objects
12     Employee employee1( "Bob", "Jones", 2875 );
13     Employee employee2( "Susan", "Baker", 3150 );
14
15     // retrieve and display employee1's monthly salary multiplied by 12
16     int monthlySalary1 = employee1.getMonthlySalary();
17     cout << "Employee 1: " << employee1.getFirstName() << " "
18         << employee1.getLastName() << "; Yearly Salary: "
19         << monthlySalary1 * 12 << endl;
20
21     // retrieve and display employee2's monthly salary multiplied by 12
22     int monthlySalary2 = employee2.getMonthlySalary();
23     cout << "Employee 2: " << employee2.getFirstName() << " "
24         << employee2.getLastName() << "; Yearly Salary: "
25         << monthlySalary2 * 12 << endl;
26
27     // give each Employee a 10% raise
28     cout << "\nIncreasing employee salaries by 10%" << endl;
29
30     employee1.setMonthlySalary( monthlySalary1 * 1.1 );
31     employee2.setMonthlySalary( monthlySalary2 * 1.1 );
32

```

Lab Exercises

Name: _____

Lab Exercise 3 — Creating an Employee Class

```
33 // retrieve and display employee1's monthly salary multiplied by 12
34 monthlySalary1 = employee1.getMonthlySalary();
35 cout << "Employee 1: " << employee1.getFirstName() << " "
36     << employee1.getLastName() << "; Yearly Salary: "
37     << monthlySalary1 * 12 << endl;
38
39 monthlySalary2 = employee2.getMonthlySalary();
40 cout << "Employee 2: " << employee2.getFirstName() << " "
41     << employee2.getLastName() << "; Yearly Salary: "
42     << monthlySalary2 * 12 << endl;
43 } // end main
```